PROCEDURE: 2027 SYSTEMS SOFTWARE DATE: 05/09/02

TITLE: Implementation of the CICS Standard Error Routine

Purpose

To establish the standards and procedures for the implementation of the CICS Standard Error Routine.

General Description

The CICS Standard Error Routine is designed to avoid CICS application ABENDs by handling unexpected return codes from CICS commands. It must be included in each program that includes "EXEC CICS" commands.

The error routine will send a message to the bottom of the terminal that initiated the CICS application. Control will return to the application program after the message has been sent.

On the production CICS system, the message will advise the end-user that an error has occurred, and that they should notify the ITD Help Desk. A printed message that gives more information about the error (in the form described for the test system; see below) will be sent concurrently to the Help Desk. The data processing analyst responsible for the CICS transaction will receive a printed report of the error on the next working day.

On the test CICS system, the message sent to the terminal is designed to give immediate diagnostic information to the programmer testing the application. The message will contain the CICS code for the command ("function") last executed, the CICS return code from that command, and the last CICS resource accessed by the program. The programmer is referred to an appendix in the CICS Application Programmer's Reference Manual for lists of CICS function codes (EIBFN) and return codes (EIBRCODE), and an explanation of the resource field (EIBRSRCE).

Implementation

To implement the CICS Standard Error Routine in a CICS program, you must include a "HANDLE CONDITION ERROR(...)" command, and LINK to the ITD-maintained Standard Error Program when the ERROR condition is invoked. This is described below, and in the examples at the end of this document. Refer to the CICS Application Programmer's Reference

Manual for more specific information about the CICS commands mentioned below.

- The Standard Error Program requires a 96-byte COMMAREA. Code the COMMAREA in Working-Storage. The first 66 bytes will be used for the CICS EXEC interface block. The last 30 bytes will be used for application-specific information, as specified by the analyst, and will appear with the message printed for the Help Desk. It is recommended that the program number be included in this information, especially for applications that execute multiple programs.

- The Standard Error Routine should be included as a separate paragraph in your program. The paragraph should include the following:
  (1) set the COMMAREA values: - move DFHEIBLK to the first 66 bytes
  - move any other useful information to the last 30 bytes.

  (2) EXEC CICS LINK to program S076J221, with the COMMAREA described above. The LINK command should include the NOHANDLE option, otherwise the application could get into a loop.

  (3) EXEC CICS RETURN, with no other parameters (or execute a paragraph that does a simple RETURN). DO NOT EXECUTE ANY OTHER CICS COMMANDS EXCEPT "RETURN" AFTER THE LINK TO S076J221, OR THE APPLICATION COULD LOOP.

- Code a HANDLE CONDITION command that includes the ERROR condition as the first CICS command executed in the program. The ERROR condition should name the paragraph that invokes the Standard Error Routine. If you use the RESP option on CICS commands, be sure to execute the paragraph that invokes the Standard Error Routine for any RESP values for which you do not test (see example 2). In this way, CICS errors that are not specifically handled by the application will result in a link to the Standard Error Routine.

```
 __________
| Examples |
|__________|
```

Example 1: An example of a CICS/COBOL II application that uses HANDLE CONDITION commands to handle expected error conditions. In this example, the application would detect the errors of a file not being open (NOTOPEN),or a record not existing in the file (NOTFND). Any other CICS errors will cause control to GO TO the paragraph 9990-STD-ERROR-ROUTINE.

```
       WORKING-STORAGE SECTION.
       ...
       01  PASS-ERR-DATA.
           05  PASS-ERR-EIB           PIC X(66).
           05  PASS-ERR-NOTE          PIC X(30) VALUE
                 'ERROR IN PROGRAM S999A999'.
       ...
       PROCEDURE DIVISION.
           EXEC CICS HANDLE CONDITION
                     NOTFND(1010-HANDLE-NOTFND)
                     NOTOPEN(9000-HANDLE-NOTOPEN)
                     ERROR(9990-STD-ERROR-ROUTINE)
           END-EXEC.
       ...
           EXEC CICS READ
                     FILE(Cxxxxxxx)
                     RIDFLD(rec-key)
                     INTO(rec-area)
                     LENGTH(rec-length)
           END-EXEC.
       ...

       1010-HANDLE-NOTFND.
       ...

       9000-HANDLE-NOTOPEN.
       ...

       9990-STD-ERROR-ROUTINE.
           MOVE DFHEIBLK TO PASS-ERR-EIB.
           EXEC CICS LINK
                     PROGRAM('S076J221')
                     COMMAREA(PASS-ERR-DATA)
                     LENGTH(96)
                     NOHANDLE
           END-EXEC.
           PERFORM 9999-RETURN-NO-TRANSID.

       9999-RETURN-NO-TRANSID.
           EXEC CICS RETURN
           END-EXEC.
           GOBACK.
```

Example 2: An example of a CICS/COBOL II application that uses the RESP option on CICS commands to handle expected error conditions. In this example, we also illustrate how the user portion of the Standard Error Routine message could contain data that might be helpful for problem resolution.

```
       WORKING-STORAGE SECTION.
       ...
       01  CICS-RESPONSE             PIC S9(8) BINARY   VALUE +0.
       ...
       01  PASS-ERR-DATA.
           05  PASS-ERR-EIB          PIC X(66).
           05  PASS-ERR-NOTE.
               10  PASS-ERR-PGM      PIC X(9)  VALUE 'S999A999 '.
```

```
              10  PASS-ERR-OTHER     PIC X(21) VALUE SPACES.
      ...

      PROCEDURE DIVISION.
          EXEC CICS HANDLE CONDITION
                    ERROR(9990-STD-ERROR-ROUTINE)
          END-EXEC.
      ...
          MOVE rec-key TO PASS-ERR-OTHER.
          EXEC CICS READ
                    FILE(Cxxxxxxx)
                    RIDFLD(rec-key)
                    INTO(rec-area)
                    LENGTH(rec-length)
                    RESP(CICS-RESPONSE)
          END-EXEC.
          EVALUATE TRUE
            WHEN CICS-RESPONSE EQUAL DFHRESP(NORMAL)
                CONTINUE
            WHEN CICS-RESPONSE EQUAL DFHRESP(NOTFND)
                PERFORM 1010-HANDLE-NOTFND
            WHEN CICS-RESPONSE EQUAL DFHRESP(NOTOPEN)
                PERFORM 9000-HANDLE-NOTOPEN
            WHEN OTHER
                PERFORM 9990-STD-ERROR-ROUTINE
          END-EVALUATE.
      ...

      1010-HANDLE-NOTFND.
      ...

      9000-HANDLE-NOTOPEN.
      ...

      9990-STD-ERROR-ROUTINE.
          MOVE DFHEIBLK TO PASS-ERR-EIB.
          EXEC CICS LINK
                    PROGRAM('S076J221')
                    COMMAREA(PASS-ERR-DATA)
                    LENGTH(96)
                    NOHANDLE
          END-EXEC.
          PERFORM 9999-RETURN-NO-TRANSID.

      9999-RETURN-NO-TRANSID.
          EXEC CICS RETURN
          END-EXEC.
          GOBACK.
```

Original issue date: 07/19/91